Innocent strategies as presheaves, and interactive equivalences for CCS

Tom Hirschowitz and Damien Pous

ICE, Reykjavik, June 2011

LAMA
Laboratoire de Mathématiques
Université de Savoie

UMR 5127

Programming languages: a technology

Claim

Research in programming languages is mainly technological.

Applies a non-formalised method, e.g.:

- Syntax.
- Quotienting by some structural congruence.
- Reduction relation to model program execution.
- Reasoning on reduction.

Goal

Find semantic frameworks for

- defining programming languages,
- reasoning about them,
- comparing them.

Leads to stupid questions like:

- What is a programming language?
- What is an observational equivalence?
- What is a compilation?

Related work

Higher-order rewriting (Nipkow, 1991)

- Models in cartesian closed 2-categories (Hirscho, 2010).
- Caveat: not yet shown to work with process calculi.

Bialgebraic semantics (Plotkin & Turi, 1997), Tile model (Gadducci & Montanari, 1996)

- Caveat: starts from syntax + labelled transition system.
- Notoriously hard to derive from structural congruence + reduction.

Outline part 1

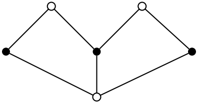
- A category of plays $\mathbb{E}$, with
  - a subcategory $\mathbb{V}$ of views,
  - a subcategory $\mathbb{W}$ of closed-world plays.
- Innocent strategies as presheaves on $\mathbb{V}$.
- Simple categorical tools $\rightsquigarrow$
  - global behaviour and
  - interaction.

Outline part 2

- Interactive equivalence of S_1 and S_2:
  - let interact with tests T,
  - observe global behaviour.
- Several notions of observation, in particular:

Fair testing = must testing.

Positions



- •'s = players,
- ∘'s = channels.
- Close to (multi-hole) active contexts in CCS:

$$\nu abc.X_1(a,c)|X_2(a,b,c)|X_3(b,c)$$

up to structural congruence.

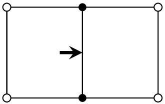
Moves and plays

- Moves: ‘elementary string diagrams’.
- Plays: certain glueings and embeddings of moves.

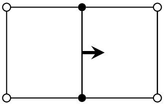
Moves from natural deduction: in/out

$$\frac{a, b \vdash P}{a, b \vdash a.P} \quad \text{and} \quad \frac{a, b \vdash P}{a, b \vdash \overline{b}.P}$$

become the following elementary string diagrams:



and



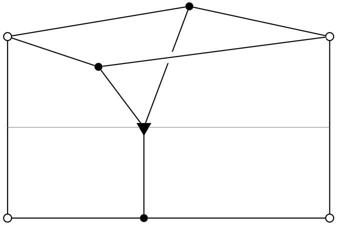
(and similarly for all 'typing' contexts).

Formally

String diagrams are presheaves over a certain category $\mathbb{C}$.

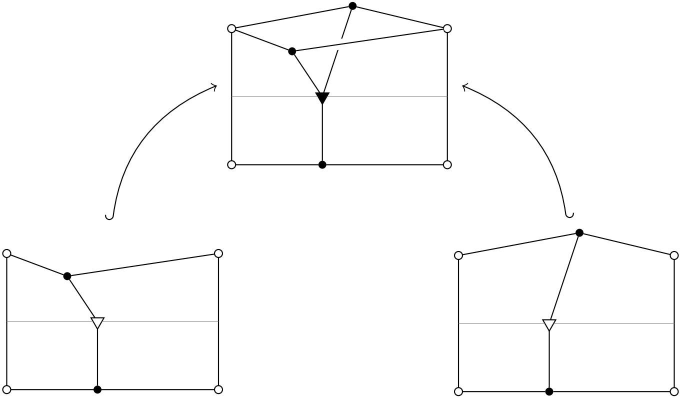
Moves from natural deduction: forking

$$\frac{a, b \vdash P \qquad a, b \vdash Q}{a, b \vdash P|Q}$$



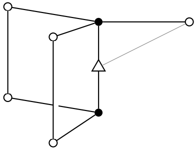
.

Views: left and right half-forking



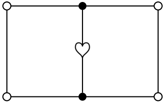
Moves from natural deduction: name creation

$$\frac{a, b, c \vdash P}{a, b \vdash \nu c.P}$$



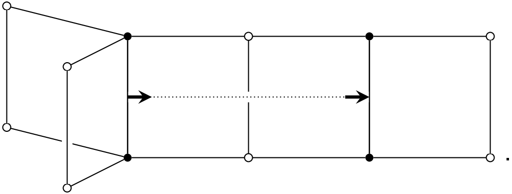
Moves from natural deduction: tick

$$\frac{a, b \vdash P}{a, b \vdash \heartsuit.P}$$



Synchronisation

$$a.P \mid \overline{a}.Q \longrightarrow P \mid Q$$

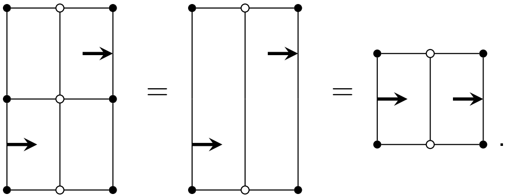


Derived from the only reduction rule, not from the LTS!

Works because positions are 'up to structural congruence'.

Plays

- Moves are plays.
- Embedding plays into larger positions yields plays.
- Piling up plays together yields plays, possibly denumerably.



The category $\mathbb{E}$ of plays

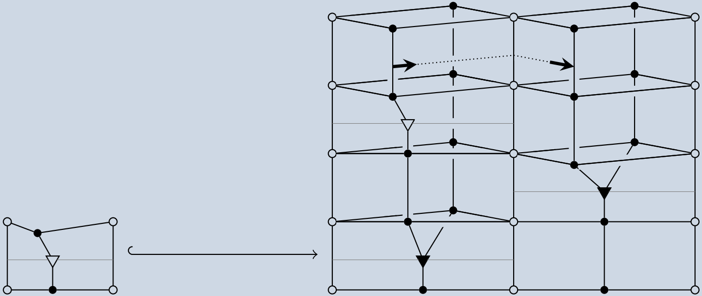
- Objects: $X \hookrightarrow U$, where X is the base position.
- Morphisms:

$$\begin{tikzcd} U \arrow[r, "k"] & V \\ X \arrow[u, hook] \arrow[r, "h"] & Y, \arrow[u, hook] \end{tikzcd}$$

i.e., embedding of base position and play.

Example

Two embeddings

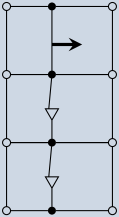


,

only one if embedding of base fixed.

The full subcategory $\mathbb{V} \hookrightarrow \mathbb{E}$ of views

Example



Strategies as functors

In standard 2-player games

Strategy $\approx$ prefix-closed set of plays.

The functorial approach

- Plays and embeddings form a poset $\mathbb{E}$.
- Strategy $\approx$ functor $S \colon \mathbb{E}^{op} \to 2$, where 2 is the poset $0 \leq 1$.
  - Play U accepted iff $S(U) = 1$.
  - For prefixes $V \hookrightarrow U$, no map $1 \to 0$, hence $S(V) = 1$ too.

Finer strategies

Naive $\mathbb{E}^{op} \to 2$.

Branching Joyal et al.: to account for branching-time subtleties.

- Switch to $\mathbb{E}^{op} \to$ Set, or FinSet...
- $S(U)$ is a set of 'states', reachable after playing U.

Sight Moves of a player should only depend on its view.

- Switch to $\mathbb{V}^{op} \to$ Set:

Questions

- How to recover the global behaviour?
- How do strategies interact?

Right Kan extension

$$\begin{tikzcd} \mathbb{V}^{op} \arrow[rr, hook, "i"] \arrow[dr, "S"'] & & \mathbb{E}^{op} \arrow[dl, "S' = \mathrm{Ran}_i(S)"] \\ & \mathrm{Set} & \end{tikzcd}$$

- End formula: $S'(U) = \int_{V \in \mathbb{V}} S(V)^{\mathbb{E}(V,U)}$.
- Global state = tuple of compatible local states.

Closed-world

- So right Kan extension yields a 'semantics' for strategies:

$$\widehat{\mathbb{V}} \to \widehat{\mathbb{E}},$$

  where $\widehat{\mathbb{V}} = \mathsf{Set}^{\mathbb{V}^{op}}$.
- But in $\mathbb{E}$ some plays
  - contain half-forkings, or
  - 'communicate with the outside'.
- Full subcat $j \colon \mathbb{W} \hookrightarrow \mathbb{E}$ of closed-world plays.
- Restriction along j, notation j^*.

Definition (Global behaviour)

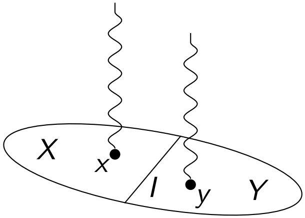
G *is the composite* $\widehat{\mathbb{V}} \xrightarrow{\mathsf{Ran}_j} \widehat{\mathbb{E}} \xrightarrow{j^*} \widehat{\mathbb{W}}$.

Interaction

- I lied: categories $\mathbb{E}_X, \mathbb{V}_X$, relative to base position X.
- If we glue two positions along certain names

$$\begin{tikzcd} I \arrow[r] \arrow[d] & Y \arrow[d] \\ X \arrow[r] & Z \arrow[ul, phantom, very near start, "\ulcorner"] \end{tikzcd}$$

- we have $\mathbb{V}_Z \cong \mathbb{V}_X + \mathbb{V}_Y$:



.

Interaction

- Because $\mathbb{V}_Z \cong \mathbb{V}_X + \mathbb{V}_Y$, copairing yields
$$\widehat{\mathbb{V}_Z} \cong \widehat{\mathbb{V}_X} \times \widehat{\mathbb{V}_Y}.$$
- For strategies S_X and S_Y on X and Y, the copairing
$$[S_X, S_Y]$$
lets S_X and S_Y play together.

Summary

- Strategies as functors $\mathbb{V}_X^{op} \to \mathrm{Set}$.
- Global behaviour $\mathrm{G}(S) =$
  right Kan extension to general plays ;
  restriction to closed-world.
- Interaction by copairing $[S_X, S_Y]$.

Now: tests.

Notion of success

Our testing semantics relies on a notion of success.

Definition

A play is successful when it contains $\heartsuit$*.*

Definition

A global behaviour $G \colon \mathbb{W}^{op} \to \mathrm{Set}$ *is*
must-successful when all its maximal executions are successful;
fair-successful when all its executions extend to successful ones.

Theorem

For all strategies S*,* $\mathrm{G}(S)$ *must-successful iff* $\mathrm{G}(S)$ *fair-successful.*

Of course there are other possibilities, e.g., only demanding that all finite executions extend to successful ones.

Tests

Definition

For a strategy S on X, a test is given by

- *a diagram $X \leftarrow I \rightarrow Y$, where*
  - *I consists of names and*
  - *Y is any position,*
- *a test strategy T.*

S passes the test iff $\mathrm{G}[S, T]$ is successful.

Definition

S and S' are equivalent when they pass the same tests.

The theorem says: must and fair are the same.

Conclusion

- A graphical notion of play, including views and closed-world plays.
- Strategies as presheaves on views.
- Interaction and global behaviour as standard categorical operations.
- $\rightsquigarrow$ testing equivalences and interpretation of CCS (not shown).

Perspectives

Short term:

- We have a translation of CCS processes into this model.
- Identify the equivalence induced by this translation.

Link with:

- graph rewriting,
- computad-based rewriting.

Longer term:

- Treat π, λ, ...
- Understand the abstract structure.
- What is a compilation?